Simple Average-Case Lower Bounds for Approximate Near-Neighbor from Isoperimetric Inequalities

Yitong Yin[*]

State Key Lab for Novel Software Technology, Nanjing University, Nanjing, China
yinyt@nju.edu.cn

Abstract

We prove an $\Omega(d/\log\frac{sw}{nd})$ lower bound for the average-case cell-probe complexity of deterministic or Las Vegas randomized algorithms solving approximate near-neighbor (ANN) problem in d-dimensional Hamming space in the cell-probe model with w-bit cells, using a table of size s. This lower bound matches the highest known worst-case cell-probe lower bounds for any static data structure problems.

This average-case cell-probe lower bound is proved in a general framework which relates the cell-probe complexity of ANN to isoperimetric inequalities in the underlying metric space. A tighter connection between ANN lower bounds and isoperimetric inequalities is established by a stronger richness lemma proved by cell-sampling techniques.

1998 ACM Subject Classification E.1 Data Structures

Keywords and phrases nearest neighbor search, approximate near-neighbor, cell-probe model, isoperimetric inequality

Digital Object Identifier 10.4230/LIPIcs.ICALP.2016.84

1 Introduction

The nearest neighbor search (NNS) problem is a fundamental problem in Computer Science. In this problem, a database $y = (y_1, y_2, \ldots, y_n)$ of n points from a metric space (X, dist) is preprocessed to a data structure, and at the query time given a query point x from the same metric space, we are asked to find the point y_i in the database which is closest to x according to the metric.

In this paper, we consider a decision and approximate version of NNS, the approximate near-neighbor (ANN) problem, where the algorithm is asked to distinguish between the two cases: (1) there is a point in the databases that is λ-close to the query point for some radius λ, or (2) all points in the database are $\gamma\lambda$-far away from the query point, where $\gamma \geq 1$ is the approximation ratio.

The complexity of nearest neighbor search has been extensively studied in the cell-probe model, a classic model for data structures. In this model, the database is encoded to a table consisting of memory cells. Upon each query, a cell-probing algorithm answers the query by making adaptive cell-probes to the table. The complexity of the problem is measured by the tradeoff between the time cost (in terms of number of cell-probes to answer a query) and the space cost (in terms of sizes of the table and cells). There is a substantial body of work on the cell-probe complexity of NNS for various metric space [6, 7, 5, 11, 8, 14, 3, 2, 16, 17, 12, 20].

[*] This work was done in part while the author was visiting the Simons Institute for the Theory of Computing. This work was partially supported by NSFC grants 61272081 and 61321491.

43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016).
Editors: Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi;
Article No. 84; pp. 84:1–84:13
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

It is widely believed that NNS suffers from the “curse of dimensionality” [10]: The problem may become intractable to solve when the dimension of the metric space becomes very high. Consider the most important example, d-dimensional Hamming space $\{0,1\}^d$ with $d \geq C \log n$ for a sufficiently large constant C. The conjecture is that NNS in this metric remains hard to solve when either approximation or randomization is allowed individually.

In a series of pioneering works [6, 5, 11, 14, 3], by a rectangle-based technique of asymmetric communication complexity known as the richness lemma [15], cell-probe lower bounds in form of $\Omega(d/\log s)$, where s stands for the number of cells in the table, were proved for deterministic approximate near-neighbor (due to Liu [14]) and randomized exact near-neighbor (due to Barkol and Rabani [5]). Such lower bound is the highest possible lower bound one can prove in the communication model. This fundamental barrier was overcome by an elegant self-reduction technique introduced in the seminal work of Pătraşcu and Thorup [18], in which the cell-probe lower bounds for deterministic ANN and randomized exact near-neighbor were improved to $\Omega(d/\log \frac{sw}{n})$, where w represents the number of bits in a cell. More recently, in a previous work of us [20], by applying the technique of Pătraşcu and Thorup to the certificates in data structures, the lower bound for deterministic ANN was further improved to $\Omega(d/\log \frac{sw}{nd})$. This last lower bound behaves differently for the polynomial space where $sw = \mathrm{poly}(n)$, near-linear space where $sw = n \cdot \mathrm{polylog}(n)$, and linear space where $sw = O(nd)$. In particular, the bound becomes $\Omega(d)$ when the space cost is strictly linear in the entropy of the database, i.e. when $sw = O(nd)$.

When both randomization and approximation are allowed, the complexity of NNS is substantially reduced. With polynomial-size tables, a $\Theta(\log\log d/\log\log\log d)$ tight bound was proved for randomized approximate NNS in d-dimensional Hamming space [7, 8]. If we only consider the decision version, the randomized ANN can be solved with $O(1)$ cell-probes on a table of polynomial size [8]. For tables of near-linear size, a technique called cell-sampling was introduced by Panigrahy *et al.* [16, 17] to prove $\Omega(\log n/\log \frac{sw}{n})$ lower bounds for randomized ANN. This was later extended to general asymmetric metrics [1].

Among these lower bounds, the randomized ANN lower bounds of Panigrahy *et al.* [16, 17] were proved explicitly for *average-case* cell-probe complexity. The significance of average-case complexity for NNS was discussed in their papers. A recent breakthrough in upper bounds [4] also attributes to solving the problem on a random database. Retrospectively, the randomized exact near-neighbor lower bounds due to the density version of richness lemma [6, 5, 11] also hold for random inputs. All these average-case lower bounds hold for Monte Carlo randomized algorithms with fixed worst-case cell-probe complexity. This leaves open an important case: the average-case cell-probe complexity for the deterministic or Las Vegas randomized algorithms for ANN, where the number of cell-probes may vary for different inputs.

1.1 Our contributions

We study the average-case cell-probe complexity of deterministic or Las Vegas randomized algorithms for the approximate near-neighbor (ANN) problem, where the number of cell-probes to answer a query may vary for different query-database pairs and the average is taken with respect to the distribution over input queries and databases.

For ANN in Hamming space $\{0,1\}^n$, the hard distribution over inputs is very natural: Every point y_i in the database $y = (y_1, y_2, \dots, y_n)$ is sampled uniformly and independently from the Hamming space $\{0,1\}^d$, and the query point x is also a point sampled uniformly and independently from $\{0,1\}^d$. According to earlier average-case lower bounds [16, 17] and the recent data-dependent LSH algorthm [4], this input distribution seems to capture the hardest case for nearest neighbor search and is also a central obstacle to overcome for efficient algorithms.

By a simple proof, we show the following lower bound for the average-case cell-probe complexity of ANN in Hamming space with this very natural input distribution.

▶ **Theorem 1.** *For $d \geq 32 \log n$ and $d < n^{o(1)}$, any deterministic or Las Vegas randomized algorithm solving (γ, λ)-approximate near-neighbor problem in d-dimensional Hamming space in the cell-probe model with w-bit cells for $w < n^{o(1)}$, using a table of size $s < 2^d$, must have expected cell-probe complexity $t = \Omega\left(\frac{d}{\gamma^2 \log \frac{sw\gamma^2}{nd}}\right)$, where the expectation is taken over both the uniform and independent input database and query and the random bits of the algorithm.*

This lower bound matches the highest known worst-case cell-probe lower bounds for *any* static data structure problems. Such lower bound was only known for polynomial evaluation [19, 13] and also worst-case deterministic ANN due to our previous work [20].

We also prove an average-case cell-probe lower bound for ANN under ℓ_∞-distance. The lower bound matches the highest known worst-case lower bound for the problem [2].

In fact, we prove these lower bounds in a unified framework that relates the average-case cell-probe complexity of ANN to isoperimetric inequalities regarding an expansion property of the metric space.

Inspired by the notions of metric expansion defined in [17], we define the following notion of expansion for metric space. Let (X, dist) be a metric space. The λ-neighborhood of a point $x \in X$, denoted as $N_\lambda(x)$ is the set of all points in X within distance λ from x. Consider a distribution μ over X. We say the λ-neighborhoods are **weakly independent** under distribution μ, if for any point $x \in X$, the measure of the λ-neighborhood $\mu(N_\lambda(x)) < \frac{\beta}{n}$ for a constant $\beta < 1$. We say the λ-neighborhoods are **(Φ, Ψ)-expanding** under distribution μ, if for any point set $A \subseteq X$ with $\mu(A) \geq \frac{1}{\Phi}$, we have $\mu(N_\lambda(A)) \geq 1 - \frac{1}{\Psi}$, where $N_\lambda(A)$ denotes the set of all points within distance λ from some point in A.

Consider the database $y = (y_1, y_2, \ldots, y_n) \in X^n$ with every point y_i sampled independently from μ, and the query $x \in X$ sampled independently from μ. We denote this input distribution as $\mu \times \mu^n$. We prove the following lower bound.

▶ **Theorem 2.** *For a metric space (X, dist), assume the followings:*

- *the $\gamma\lambda$-neighborhoods are weakly independent under distribution μ;*
- *the λ-neighborhoods are (Φ, Ψ)-expanding under distribution μ.*

Then any deterministic or Las Vegas randomized algorithm solving (γ, λ)-approximate near-neighbor problem in (X, dist) in the cell-probe model with w-bit cells, using a table of size s, must have expected cell-probe complexity

$$t = \Omega\left(\frac{\log \Phi}{\log \frac{sw}{n \log \Psi}}\right) \quad \text{or} \quad t = \Omega\left(\frac{n \log \Psi}{w + \log s}\right)$$

under input distribution $\mu \times \mu^n$.

The key step to prove such a theorem is a stronger version of the richness lemma that we prove in Section 3. The proof of this stronger richness lemma uses an idea called "cell-sampling" introduced by Panigrahy *et al.* [17] and later refined by Larsen [13]. This new richness lemma as well as this connection between the rectangle-based techniques (such as the richness lemma) and information-theory-based techniques (such as cell-sampling) are of interests by themselves.

2 Preliminary

Let (X, dist) be a metric space. Let $\gamma \geq 1$ and $\lambda \geq 0$. The **(γ, λ)-approximate near-neighbor problem** $(\gamma, \lambda)\text{-}\mathsf{ANN}_X^n$ is defined as follows: A database $y = (y_1, y_2, \ldots, y_n) \in X^n$ of n points from X is preprocessed and stored as a data structure. Upon each query $x \in X$, by accessing the data structure we want to distinguish between the following two cases: (1) there is a point y_i in the database such that $\mathrm{dist}(x, z) \leq \lambda$; (2) for all points y_i in the database we have $\mathrm{dist}(x, z) > \gamma\lambda$. For all other cases the answer can be arbitrary.

More abstractly, given a universe X of queries and a universe Y of all databases, a **data structure problem** is a function $f : X \times Y \to Z$ that maps every pair of **query** $x \in X$ and **database** $y \in Y$ to an answer $f(x, y) \in Z$. In our example of $(\gamma, \lambda)\text{-}\mathsf{ANN}_X^n$, the query universe is the metric space X, the database universe is the set $Y = X^n$ of all tuples of n points from X, and f maps each query $x \in X$ and database $y \in Y$ to an Boolean answer: $f(x, y) = 0$ if there is a λ-near neighbor of x in the database y; $f(x, y) = 1$ if no points in the database y is a $\gamma\lambda$-near neighbor of x; and $f(x, y)$ can be arbitrary if otherwise. Note that due to a technical reason, we usually use 1 to indicate the "no near-neighbor" case.

Given a data structure problem $f : X \times Y \to Z$, a code $T : Y \to \Sigma^s$ with alphabet $\Sigma = \{0, 1\}^w$ encodes every database $y \in Y$ to a **table** T_y of s **cells** with each cell storing a word of w bits. We use $[s] = \{1, 2, \ldots, s\}$ to denote the set of indices of cells. For each $i \in [s]$, we use $T_y[i]$ to denote the content of the i-th cell of table T_y; and for $S \subseteq [s]$, we write $T_y[S] = (T_y[i])_{i \in S}$ for the tuple of the contents of the cells in S. Upon each query $x \in X$, a **cell-probing algorithm** adaptive retrieves the contents of the cells in the table T_y (which is called **cell-probes**) and outputs the answer $f(x, y)$ at last. Being adaptive means that the cell-probing algorithm is actually a decision tree: In each round of cell-probing the address of the cell to probe next is determined by the query x as well as the contents of the cells probed in previous rounds. Together, this pair of code and decision tree is called a **cell-probing scheme**.

For randomized cell-probing schemes, the cell-probing algorithm takes a sequence of random bits as its internal random coin. In this paper we consider only deterministic or Las Vegas randomized cell-probing algorithms, therefore the algorithm is guaranteed to output a correct answer when it terminates.

When a cell-probing scheme is fixed, the size s of the table as well as the length w of each cell are fixed. These two parameters together give the space complexity. And the number of cell-probes may vary for each pair of inputs (x, y) or may be a random variable if the algorithm is randomized. Given a distribution $\mathcal{D}$ over $X \times Y$, the **average-case cell-probe complexity** for the cell-probing scheme is given by the expected number of cell-probes to answer $f(\boldsymbol{x}, \boldsymbol{y})$ for a $(\boldsymbol{x}, \boldsymbol{y})$ sampled from $\mathcal{D}$, where the expectation is taken over both the input distribution $\mathcal{D}$ and the internal random bits of the cell-probing algorithm.

3 A richness lemma for average-case cell-probe complexity

The richness lemma (or the rectangle method) introduced in [15] is a classic tool for proving cell-probe lower bounds. A data structure problem $f : X \times Y \to \{0, 1\}$ is a natural communication problem, and a cell-probing scheme can be interpreted as a communication protocol between the cell-probing algorithm and the table, with cell-probes as communications.

Given a distribution $\mathcal{D}$ over $X \times Y$, a data structure problem $f : X \times Y \to \{0, 1\}$ is **α-dense** under distribution $\mathcal{D}$ if $\mathbb{E}_{\mathcal{D}}[f(\boldsymbol{x}, \boldsymbol{y})] \geq \alpha$. A combinatorial rectangle $A \times B$ for $A \subseteq X$ and $B \subseteq Y$ is a monochromatic 1-rectangle in f if $f(x, y) = 1$ for all $(x, y) \in A \times B$.

The richness lemma states that if a problem f is dense enough (i.e. being rich in 1's) and is easy to solve by communication, then f contains large monochromatic 1-rectangles. Specifically, if an α-dense problem f can be solved by Alice sending a bits and Bob sending b bits in total, then f contains a monochromatic 1-rectangle of size $\alpha \cdot 2^{-O(a)} \times \alpha \cdot 2^{-O(a+b)}$ in the uniform measure. In the cell-probe model with w-bit cells, tables of size s and cell-probe complexity t, it means the monochromatic 1-rectangle is of size $\alpha \cdot 2^{-O(t \log s)} \times \alpha \cdot 2^{-O(t \log s + tw)}$. The cell-probe lower bounds can then be proved by refuting such large 1-rectangles for specific data structure problems f.

We prove the following richness lemma for average-case cell-probe complexity.

▶ **Lemma 3.** *Let μ, ν be distributions over X and Y respectively, and let $f : X \times Y \to \{0,1\}$ be α-dense under the product distribution $\mu \times \nu$. If there is a deterministic or randomized Las Vegas cell-probing scheme solving f on a table of s cells, each cell containing w bits, with expected t cell-probes under input distribution $\mu \times \nu$, then for any $\Delta \in \left[32t/\alpha^2, s\right]$, there is a monochromatic 1-rectangle $A \times B \subseteq X \times Y$ in f such that $\mu(A) \geq \alpha \cdot \left(\frac{\Delta}{s}\right)^{O(t/\alpha^2)}$ and $\nu(B) \geq \alpha \cdot 2^{-O(\Delta \ln \frac{s}{\Delta} + \Delta w)}$.*

Compared to the classic richness lemma, this new lemma has the following advantages:

- It holds for average-case cell-probe complexity.
- It gives stronger result even restricted to worst-case complexity. The newly introduced parameter Δ should not be confused as an overhead caused by the average-case complexity argument, rather, it strengthens the result even for the worst-case lower bounds. When $\Delta = t$ it gives the bound in the classic richness lemma.
- The lemma claims the existence of a *family* of rectangles parameterized by Δ, therefore to prove a cell-probe lower bound it is enough to refute any one rectangle from this family. As we will see, this gives us a power to prove the highest lower bounds (even for the worst case) known to any static data structure problems.

The proof of this lemma uses an argument called "cell-sampling" introduced by Panigrahy *et al.* [16, 17] for approximate nearest neighbor search and later refined by Larsen [13] for polynomial evaluation. Our proof is greatly influenced by Larsen's approach.

The rest of this section is dedicated to the proof of this lemma.

3.1 Proof of the average-case richness lemma (Lemma 3)

By fixing random bits, it is sufficient to consider only deterministic cell-probing algorithms.

The high level idea of the proof is simple. Fix a table T_y. A procedure called the "cell-sampling procedure" chooses the subset Γ of Δ many cells that resolve the maximum amount of positive queries. This associates each database y to a string $\omega = (\Gamma, T_y[\Gamma])$, which we call a **certificate**, where $T_y[\Gamma] = (T_y[i])_{i \in \Gamma}$ represent the contents of the cells in Γ. Due to the nature of the cell-probing algorithm, once the certificate is fixed, the set of queries it can resolve is fixed. We also observe that if the density of 1's in the problem f is $\Omega(1)$, then there is a $\Omega(1)$-fraction of good databases y such that amount of positive queries resolved by the certificate ω constructed by the cell-sampling procedure is at least an $\left(\frac{\Delta}{s}\right)^{O(t)}$-fraction of all queries. On the other hand, since $\omega \in \binom{[s]}{\Delta} \times \{0,1\}^{\Delta w}$ there are at most $\binom{s}{\Delta} 2^{\Delta w} = 2^{O(\Delta \ln \frac{s}{\Delta} + \Delta w)}$ many certificates ω. Therefore, at least $2^{-O(\Delta \ln \frac{s}{\Delta} + \Delta w)}$-fraction of good databases (which is at least $2^{-O(\Delta \ln \frac{s}{\Delta} + \Delta w)}$-fraction of all databases) are associated with the same ω. Pick this popular certificate ω, the positive queries that ω resolves together with the good databases that ω is associated with form the large monochromatic 1-rectangle.

Now we proceed to the formal parts of the proof. Given a database $y \in Y$, let $X_y^+ = \{x \in X \mid f(x, y) = 1\}$ denote the set of positive queries on y. We use $\mu_y^+ = \mu_{X_y^+}$ to denote the distribution induced by μ on X_y^+.

Let $P_{xy} \subseteq [s]$ denote the set of cells probed by the algorithm to resolve query x on database y. Fix a database $y \in Y$. Let $\Gamma \subseteq [s]$ be a subset of cells. We say a query $x \in X$ is resolved by Γ if x can be resolved by probing only cells in Γ on the table storing database y, i.e. if $P_{xy} \subseteq \Gamma$. We denote by

$$X_y^+(\Gamma) = \{x \in X_y^+ \mid P_{xy} \subseteq \Gamma\}$$

the set of positive queries resolved by Γ on database y. Assume two databases y and y' are *indistinguishable* over Γ: meaning that for the tables T_y and $T_{y'}$ storing y and y' respectively, the cell contents $T_y[i] = T_{y'}[i]$ for all $i \in \Gamma$. Then due to the determinism of the cell-probing algorithm, we have $X_y^+(\Gamma) = X_{y'}^+(\Gamma)$, i.e. Γ resolve the same set of positive queries on both databases.

The cell-sampling procedure

Fix a database $y \in Y$ and any $\Delta \in \left[32t/\alpha^2, s\right]$. Suppose we have a *cell-sampling procedure* which does the following: The procedure deterministically[1] chooses a unique $\Gamma \subseteq [s]$ such that $|\Gamma| = \Delta$ and the measure $\mu(X_y^+(\Gamma))$ of positive queries resolved by Γ is maximized (and if there are more than one such Γ, the procedure chooses an arbitrary one of them). We use Γ_y^* to denote this set of cells chosen by the cell-sampling procedure. We also denote by $X_y^* = X_y^+(\Gamma_y^*)$ the set of positive queries resolved by this chosen set of cells.

On each database y, the cell-sampling procedure chooses for us the most informative set Γ of cells of size $|\Gamma| = \Delta$ that resolve the maximum amount of positive queries. We use $\omega_y = (\Gamma_y^*, T_y[\Gamma_y^*])$ to denote the contents (along with addresses) of the cells chosen by the cell-sampling procedure for database y. We call such ω_y a **certificate** chosen by the cell-sampling procedure for y.

Let y and y' be two databases. A simple observation is that if two databases y and y' have the same certificate $\omega_y = \omega_{y'}$ chosen by the cell-sampling procedure, then the respective sets $X_y^*, X_{y'}^*$ of positive queries resolved on the certificate are going to be the same as well.

▶ **Proposition 4.** *For any databases $y, y' \in Y$, if $\omega_y = \omega_{y'}$ then $X_y^* = X_{y'}^*$.*

Let $\tau(x, y) = |P(x, y)|$ denote the number of cell-probes to resolve query x on database y. By the assumption of the lemma, $\mathbb{E}_{\mu\times\nu}[\tau(\boldsymbol{x}, \boldsymbol{y})] \leq t$ for the inputs $(\boldsymbol{x}, \boldsymbol{y})$ sampled from the product distribution $\mu \times \nu$. We claim that there are many "good" columns (databases) with high density of 1's and low average cell-probe costs.

▶ **Claim 5.** *There is a collection $Y_{\mathsf{good}} \subseteq Y$ of substantial amount of good databases, such that $\nu(Y_{\mathsf{good}}) \geq \frac{\alpha}{4}$ and for every $y \in Y_{\mathsf{good}}$, the followings are true:*

- *the amount of positive queries is large: $\mu(X_y^+) \geq \frac{\alpha}{2}$;*
- *the average cell-probe complexity* among positive queries *is bounded:*

$$\mathbb{E}_{\boldsymbol{x}\sim\mu_y^+}[\tau(\boldsymbol{x}, y)] \leq \frac{8t}{\alpha^2}.$$

[1] Being deterministic here means that the chosen set Γ_y^* is a function of y.

Proof. The claim is proved by a series of averaging principles. First consider $Y_{\mathsf{dense}} = \{y \in Y \mid \mu(X_y^+) \geq \frac{\alpha}{2}\}$ the set of databases with at least $\frac{\alpha}{2}$-density of positive queries. By the averaging principle, we have $\nu(Y_{\mathsf{dense}}) \geq \alpha/2$. Since $\mathbb{E}[\tau(\boldsymbol{x}, \boldsymbol{y})] \geq \nu(Y_{\mathsf{dense}})\mathbb{E}[\tau(\boldsymbol{x}, \boldsymbol{y}) \mid y \in Y_{\mathsf{dense}}]$, we have $\mathbb{E}_{\mu\times\nu_{\mathsf{dense}}}[\tau(\boldsymbol{x}, \boldsymbol{y})] \leq \frac{2t}{\alpha}$, where $\nu_{\mathsf{dense}} = \nu_{Y_{\mathsf{dense}}}$ is the distribution induced by ν on Y_{dense}. We then construct $Y_{\mathsf{good}} \subseteq Y_{\mathsf{dense}}$ as the set of $y \in Y_{\mathsf{dense}}$ with average cell-probe complexity bounded as $\mathbb{E}_{\boldsymbol{x}\sim\mu}[\tau(\boldsymbol{x}, y)] \leq \frac{4t}{\alpha}$. By Markov inequality $\nu_{\mathsf{dense}}(Y_{\mathsf{good}}) \geq \frac{1}{2}$ and hence $\nu(Y_{\mathsf{good}}) \geq \frac{\alpha}{4}$. Note that $\mathbb{E}_{\boldsymbol{x}\sim\mu}[\tau(\boldsymbol{x}, y)] \geq \mathbb{E}_{\boldsymbol{x}\sim\mu_y^+}[\tau(\boldsymbol{x}, y)]\mu(X_y^+)$. We have $\mathbb{E}_{\boldsymbol{x}\sim\mu_y^+}[\tau(\boldsymbol{x}, y)] \leq \mathbb{E}_{\boldsymbol{x}\sim\mu}[\tau(\boldsymbol{x}, y)]/\mu(X_y^+) \leq \frac{8t}{\alpha^2}$ for all $y \in Y_{\mathsf{good}}$. ◀

For the rest, we consider only these good databases. Fix any $\Delta \in \left[32t/\alpha^2, s\right]$. We claim that for every good database $y \in Y_{\mathsf{good}}$, the cell-sampling procedure always picks a subset $\Gamma_y^* \subseteq [s]$ of Δ many cells, which can resolve a substantial amount of positive queries:

▶ **Claim 6.** *For every $y \in Y_{\mathsf{good}}$, it holds that $\mu(X_y^*) \geq \frac{\alpha}{4}\left(\frac{\Delta}{2s}\right)^{8t/\alpha^2}$.*

Proof. Fix any good database $y \in Y_{\mathsf{good}}$. We only need to prove there exists a $\Gamma \subseteq [s]$ with $|\Gamma| = \Delta$ that resolve positive queries $\mu(X_y^+(\Gamma)) \geq \frac{\alpha}{4}\left(\frac{\Delta}{2s}\right)^{8t/\alpha^2}$. The claims follows immediately.

We construct a hypergraph $\mathcal{H} \subseteq 2^{[s]}$ with vertex set $[s]$ as $\mathcal{H} = \{P_{xy} \mid x \in X_y^+\}$, so that each positive queries $x \in X_y^+$ on database y is associated (many-to-one) to a hyperedge $e \in \mathcal{H}$ such that $e = P_{xy}$ is precisely the set of cells probed by the cell-probing algorithm to resolve query x on database y.

We also define a measure $\tilde{\mu}$ over hyperedges $e \in \mathcal{H}$ as the total measure (in μ_y^+) of the positive queries x associated to e. Formally, for every $e \in \mathcal{H}$,

$$\tilde{\mu}(e) = \sum_{x\in X_y^+:P_{xy}=e} \mu_y^+(x).$$

Since $\sum_{e\in\mathcal{H}} \tilde{\mu}(e) = \sum_{x\in X_y^+} \mu_y^+(x) = 1$, this $\tilde{\mu}$ is a well-defined probability distribution over hyperedges in $\mathcal{H}$. Moreover, recalling that $\tau(x, y) = |P_{xy}|$, the the average size of hyperedges

$$\mathbb{E}_{\boldsymbol{e}\sim\tilde{\mu}}[|\boldsymbol{e}|] = \mathbb{E}_{\boldsymbol{x}\sim\mu_y^+}[\tau(\boldsymbol{x}, y)] \leq \frac{8t}{\alpha^2}.$$

By the probabilistic method (whose proof is in the full paper [21]), there must exist a $\Gamma \subseteq [s]$ of size $|\Gamma| = \Delta$, such that the sub-hypergraph $\mathcal{H}_\Gamma$ induced by Γ has

$$\tilde{\mu}(\mathcal{H}_\Gamma) \geq \frac{1}{2}\left(\frac{\Delta}{2s}\right)^{8t/\alpha^2}.$$

By our construction of $\mathcal{H}$, the positive queries associated (many-to-one) to the hyperedges in the induced sub-hypergraph $\mathcal{H}_\Gamma = \{P_{xy} \mid x \in X_y^+ \wedge P_{xy} \subseteq \Gamma\}$ are precisely those positive queries in $X_y^+(\Gamma) = \{x \in X_y^+ \mid P_{xy} \subseteq \Gamma\}$. Therefore,

$$\mu_y^+(X_y^+(\Gamma)) = \sum_{x\in X_y^+, P_{xy}\subseteq\Gamma} \mu_y^+(x) = \tilde{\mu}(\mathcal{H}_\Gamma) \geq \frac{1}{2}\left(\frac{\Delta}{2s}\right)^{8t/\alpha^2}.$$

Recall that $\mu(X_y^+) \geq \frac{\alpha}{2}$ for every $y \in Y_{\mathsf{good}}$. And since $X_y^+(\Gamma) \subseteq X_y^+$, we have

$$\mu(X_y^+(\Gamma)) = \mu_y^+(X_y^+(\Gamma))\mu(X_y^+) \geq \frac{\alpha}{4}\left(\frac{\Delta}{2s}\right)^{8t/\alpha^2}.$$

The claim is proved. ◀

Recall that the certificate $\omega_y = (\Gamma_y^*, T_y[\Gamma_y^*])$ is constructed by the cell-sampling procedure for database y. For every possible assignment $\omega \in \binom{[s]}{\Delta} \times \{0,1\}^{\Delta w}$ of certificate, let Y_ω denote the set of good databases $y \in Y_{\mathsf{good}}$ with this certificate $\omega_y = \omega$. Due to the determinism of the cell-sampling procedure, this classifies the Y_{good} into at most $\binom{s}{\Delta}2^{\Delta w}$ many disjointed subclasses Y_ω. Recall that $\nu(Y_{\mathsf{good}}) \geq \frac{\alpha}{4}$. By the averaging principle, the following proposition is natural.

▶ **Proposition 7.** *There exists a certificate $\omega \in \binom{[s]}{\Delta} \times \{0,1\}^{\Delta w}$, denoted as ω^*, such that*

$$\nu(Y_{\omega^*}) \geq \frac{\alpha}{4\binom{s}{\Delta}2^{\Delta w}}.$$

On the other hand, fixed any ω, since all databases $y \in Y_\omega$ have the same ω_y^*, by Proposition 4 they must have the same X_y^*. We can abuse the notation and write $X_\omega = X_y^*$ for all $y \in Y_\omega$.

Now we let $A = X_{\omega^*}$ and $B = Y_{\omega^*}$, where ω^* satisfies Proposition 7. Due to Claim 6 and Proposition 7, we have

$$\mu(A) \geq \frac{\alpha}{4}\left(\frac{\Delta}{2s}\right)^{8t/\alpha^2} = \alpha \cdot \left(\frac{\Delta}{s}\right)^{O(t/\alpha^2)} \quad \text{and} \quad \nu(B) \geq \frac{\alpha}{4\binom{s}{\Delta}2^{\Delta w}} = \alpha \cdot 2^{-O\left(\Delta \ln \frac{s}{\Delta} + \Delta w\right)}.$$

Note for every $y \in B = Y_{\omega^*}$, the $A = X_{\omega^*} = X_y^+(\Gamma_y^*)$ is a set of positive queries on database y, thus $A \times B$ is a monochromatic 1-rectangle in f. This finishes the proof of Lemma 3.

4 Rectangles in conjunction problems

Many natural data structure problems can be expressed as a conjunction of point-wise relations between the query point and database points. Consider data structure problem $f : X \times Y \to \{0,1\}$. Let $Y = \mathcal{Y}^n$, so that each database $y \in Y$ is a tuple $y = (y_1, y_2, \ldots, y_n)$ of n points from $\mathcal{Y}$. A **point-wise function** $g : X \times \mathcal{Y} \to \{0,1\}$ is given. The data structure problem f is defined as the conjunction of these subproblems:

$$\forall x \in X, \forall y = (y_1, y_2, \ldots, y_n) \in Y, \quad f(x,y) = \bigwedge_{i=1}^{n} g(x, y_i)\,.$$

Many natural data structure problems can be defined in this way, for example:

- Membership query: $X = \mathcal{Y}$ is a finite domain. The point-wise function $g(\cdot,\cdot)$ is $\neq$ that indicates whether the two points are unequal.
- (γ,λ)-approximate near-neighbor (γ,λ)-ANN_X^n: $X = \mathcal{Y}$ is a metric space with distance $\mathrm{dist}(\cdot,\cdot)$. The point-wise function g is defined as: for $x, z \in X$, $g(x,z) = 1$ if $\mathrm{dist}(x,z) > \gamma\lambda$, or $g(x,z) = 0$ if $\mathrm{dist}(x,z) \leq \lambda$. The function value can arbitrary for all other cases.
- Partial match $\mathsf{PM}_\Sigma^{d,n}$: Σ is an alphabet, $\mathcal{Y} = \Sigma^d$ and $X = (\Sigma \cup \{\star\})^d$. The point-wise function g is defined as: for $x \in X$ and $z \in \mathcal{Y}$, $g(x,z) = 1$ if there is an $i \in [d]$ such that $x_i \notin \{\star, z_i\}$, or $g(x,z) = 0$ if otherwise.

We show that refuting the large rectangles in the point-wise function g can give us lower bounds for the conjunction problem f.

Let μ, ν be distributions over X and $\mathcal{Y}$ respectively, and let ν^n be the product distribution on $Y = \mathcal{Y}^n$. Let $g : X \times \mathcal{Y} \to \{0,1\}$ be a point-wise function and $f : X \times Y \to \{0,1\}$ a data structure problem defined by the conjunction of g as above.

▶ **Lemma 8.** *For f, g, μ, ν defined as above, assume that there is a deterministic or randomized Las Vegas cell-probing scheme solving f on a table of s cells, each cell containing w bits, with expected t cell-probes under input distribution $\mu \times \nu^n$. If the followings are true:*

- *the density of 0's in g is at most $\frac{\beta}{n}$ under distribution $\mu \times \nu$ for some constant $\beta < 1$;*
- *g does not contain monochromatic 1-rectangle of measure at least $\frac{1}{\Phi} \times \frac{1}{\Psi}$ under distribution $\mu \times \nu$;*

then

$$\left(\frac{sw}{n\log\Psi}\right)^{O(t)} \geq \Phi \quad \text{or} \quad t = \Omega\left(\frac{n\log\Psi}{w+\log s}\right).$$

Proof. By union bound, the density of 0's in f under distribution $\mu \times \nu^n$ is:

$$\Pr_{\substack{x\sim\mu \\ y=(y_1,\ldots,y_n)\sim\nu^n}}\left[\bigwedge_{i=1}^{n} g(x,y_i)=0\right] \leq n\cdot\Pr_{\substack{x\sim\mu\\ z\sim\nu}}[g(x,z)=0] \leq n\cdot\frac{\beta}{n}=\beta.$$

By Lemma 3, the $\Omega(1)$-density of 1's in f and the assumption of existing a cell-probing scheme with parameters s, w and t, altogether imply that for any $4t \leq \Delta \leq s$, f has a monochromatic 1-rectangle $A \times B$ such that

$$\mu(A) \geq \left(\frac{\Delta}{s}\right)^{c_1 t} \quad \text{and} \quad \nu^n(B) \geq 2^{-c_2\Delta(\ln\frac{s}{\Delta}+w)}, \tag{1}$$

for some constants $c_1, c_2 > 0$ depending only on β.

Let $C \subset \mathcal{Y}$ be the largest set of columns in g to form a 1-rectangle with A. Formally,

$$C = \{z \in \mathcal{Y} \mid \forall x \in A, g(x,z)=1\}.$$

Clearly, for any monochromatic 1-rectangle $A \times D$ in g, we must have $D \subseteq C$. By definition of f as a conjunction of g, it must hold that for all $y = (y_1, y_2, \ldots, y_n) \in B$, none of $y_i \in y$ has $g(x, y_i) = 0$ for any $x \in A$, which means $B \subseteq C^n$, and hence

$$\nu^n(B) \leq \nu^n(C^n) = \nu(C)^n.$$

Recall that $A \times C$ is monochromatic 1-rectangle in g. Due to the assumption of the lemma, either $\mu(A) < \frac{1}{\Phi}$ or $\nu(C) < \frac{1}{\Psi}$. Therefore, either $\mu(A) < \frac{1}{\Phi}$ or $\nu^n(B) < \frac{1}{\Psi^n}$.

We can always choose a Δ such that $\Delta = O\left(\frac{n\log\Psi}{w}\right)$ and $\Delta = \Omega\left(\frac{n\log\Psi}{w+\log s}\right)$ to satisfy

$$2^{-c_2\Delta(\ln\frac{s}{\Delta}+w)} > \frac{1}{\Psi^n}.$$

If such Δ is less than $32t/(1-\beta)^2$, then we immediately have a lower bound

$$t = \Omega\left(\frac{n\log\Psi}{w+\log s}\right).$$

Otherwise, due to (1), $A \times B$ is monochromatic 1-rectangle in f with $\nu^n(B) > \frac{1}{\Psi^n}$, therefore it must hold that $\mu(A) < \frac{1}{\Phi}$, which by (1) gives us

$$\frac{1}{\Phi} > \mu(A) \geq \left(\frac{\Delta}{s}\right)^{O(t)} = \left(\frac{n\log\Psi}{sw}\right)^{O(t)},$$

which gives the lower bound

$$\left(\frac{sw}{n\log\Psi}\right)^{O(t)} \geq \Phi. \quad ◀$$

5 Isoperimetry and ANN lower bounds

Given a metric space X with distance $\mathrm{dist}(\cdot,\cdot)$ and $\lambda \geq 0$, we say that two points $x, x' \in X$ are λ-close if $\mathrm{dist}(x,x') \leq \lambda$, and λ-far if otherwise. The λ-neighborhood of a point $x \in X$, denoted by $N_\lambda(x)$, is the set of all points from X which are λ-close to x. Given a point set $A \subseteq X$, we define $N_\lambda(A) = \bigcup_{x\in A} N_\lambda(x)$ to be the set of all points which are λ-close to some point in A.

In [17], a natural notion of metric expansion was introduced.

▶ **Definition 9** (metric expansion [17])**.** Let X be a metric space and μ a probability distribution over X. Fix any radius $\lambda > 0$. Define

$$\Phi(\delta) \triangleq \min_{A\subset X, \mu(A)\leq\delta} \frac{\mu(N_\lambda(A))}{\mu(A)}.$$

The expansion Φ of the λ-neighborhoods in X under distribution μ is defined as the largest k such that for all $\delta \leq \frac{1}{2k}$, $\Phi(\delta) \geq k$.

We now introduce a more refined definition of metric expansion using two parameters Φ and Ψ.

▶ **Definition 10** ((Φ,Ψ)-expanding)**.** Let X be a metric space and μ a probability distribution over X. The λ-neighborhoods in X are **(Φ,Ψ)-expanding** under distributions μ if we have $\mu(N_\lambda(A)) \geq 1 - 1/\Psi$ for any $A \subseteq X$ that $\mu(A) \geq 1/\Phi$.

The metric expansion defined in [17] is actually a special case of (Φ,Ψ)-expanding: The expansion of λ-neighborhoods in a metric space X is Φ means the λ-neighborhoods are $(\Phi, 2)$-expanding. The notion of (Φ,Ψ)-expanding allows us to describe a more extremal expanding situation in metric space: The expanding of λ-neighborhoods does not stop at measure 1/2, rather, it can go all the way to be very close to measure 1. This generality may support higher lower bounds for approximate near-neighbor.

Given a radius $\lambda > 0$ and an approximation ratio $\gamma > 1$, recall that the (γ,λ)-approximate near neighbor problem (γ,λ)-ANN_X^n can be defined as a conjunction $f(x,y) = \bigwedge_i g(x,y_i)$ of point-wise function $g : X \times X \to \{0,1\}$ where $g(x,z) = 0$ if x is λ-close to z; $g(x,z) = 1$ if x is $\gamma\lambda$-far from z; and $g(x,z)$ is arbitrary for all other cases. Observe that g is actually (γ,λ)-ANN_X^1, the point-to-point version of the (γ,λ)-approximate near neighbor.

The following proposition gives an intrinsic connection between the expansion of metric space and size of monochromatic rectangle in the point-wise near-neighbor relation.

▶ **Proposition 11.** *If the λ-neighborhoods in X are (Φ,Ψ)-expanding under distribution μ, then the function g defined as above does not contain a monochromatic 1-rectangle of measure $\geq \frac{1}{\Phi} \times \frac{1.01}{\Psi}$ under distribution $\mu \times \mu$.*

Proof. Since the λ-neighborhoods in X are (Φ,Ψ)-expanding, for any $A \subseteq X$ with $\mu(A) \geq \frac{1}{\Phi}$, we have $\mu(N_\lambda(A)) \geq 1 - \frac{1}{\Psi}$. And by definition of g, for any monochromatic $A \times B$, it must hold that $B \cap N_\lambda(A) = \emptyset$, i.e. $B \subseteq X \setminus N_\lambda(A)$. Therefore, either $\mu(A) < \frac{1}{\Phi}$, or $\mu(B) = 1 - \mu(N_\lambda(A)) \leq \frac{1}{\Psi} < \frac{1.01}{\Psi}$. ◀

The above proposition together with Lemma 8 immediately gives us the following corollary which reduces lower bounds for near-neighbor problems to the isoperimetric inequalities.

▶ **Corollary 12.** *Let μ be a distribution over a metric space X. Let $\lambda > 0$ and $\gamma \geq 1$. Assume that there is a deterministic or randomized Las Vegas cell-probing scheme solving (γ,λ)-ANN_X^n on a table of s cells, each cell containing w bits, with expected t cell-probes under input distribution $\mu \times \mu^n$. If the followings are true:*

- $\mathbb{E}_{x\sim\mu}\left[\mu(N_{\gamma\lambda}(x))\right] \leq \frac{\beta}{n}$ *for a constant* $\beta < 1$*;*
- *the* λ*-neighborhoods in* X *are* (Φ, Ψ)*-expanding under distribution* μ*;*

then

$$\left(\frac{sw}{n\log\Psi}\right)^{O(t)} \geq \Phi \quad \text{or} \quad t = \Omega\left(\frac{n\log\Psi}{w+\log s}\right).$$

▶ Remark. In [17], a lower bound for (γ,λ)-ANN^n_X was proved with the following form:

$$\left(\frac{swt}{n}\right)^{t} \geq \Phi.$$

In our Corollary 12, unless the cell-size w is unrealistically large to be comparable to n, the corollary always gives the first lower bound

$$\left(\frac{sw}{n\log\Psi}\right)^{O(t)} \geq \Phi.$$

This strictly improves the lower bound in [17]. For example, when the metric space is $\left(2^{\Theta(d)}, 2^{\Theta(d)}\right)$-expanding, this would give us a lower bound $t = \Omega\left(\frac{d}{\log\frac{sw}{nd}}\right)$, which in particular, when the space is linear ($sw = O(nd)$), becomes $t = \Omega(d)$.

5.1 Lower bound for ANN in Hamming space

Let $X = \{0,1\}^d$ be the Hamming space with Hamming distance $\mathrm{dist}(\cdot,\cdot)$. Recall that $N_\lambda(x)$ represents the λ-neighborhood around x, in this case, the Hamming ball of radius λ centered at x; and for a set $A \subset X$, the $N_\lambda(A)$ is the set of all points within distance λ to any point in A. For any $0 \leq r \leq d$ $B(r) = |N_r(\bar{0})|$ denote the volume of Hamming ball of radius r, where $\bar{0} \in \{0,1\}^d$ is the zero vector. Obviously $B(r) = \sum_{k\leq r}\binom{d}{k}$.

The following isoperimetric inequality of Harper is well known.

▶ **Lemma 13** (Harper's theorem [9]). *Let* $X = \{0,1\}^d$ *be the* d*-dimensional Hamming space. For* $A \subset X$*, let* r *be such that* $|A| \geq B(r)$*. Then for every* $\lambda > 0$*,* $|N_\lambda(A)| \geq B(r+\lambda)$*.*

In words, Hamming balls have the worst vertex expansion.

For $0 < r < \frac{d}{2}$, the following upper bound for the volume of Hamming ball is well known:

$$2^{(1-o(1))dH(r/d)} \leq \binom{d}{r} \leq B(r) \leq 2^{dH(r/d)},$$

where $H(x) = -x\log_2 x - (1-x)\log_2(1-x)$ is the Boolean entropy function.

Consider the Hamming (γ,λ)-approximate near-neighbor problem (γ,λ)-ANN^n_X. The hard distribution for this problem is just the uniform and independent distribution: For the database $y = (y_1, y_2, \ldots, y_n) \in X^n$, each database point y_i is sampled uniformly and independently from $X = \{0,1\}^n$; and the query point x is sampled uniformly and independently from X.

▶ **Theorem 14.** *Let* $d \geq 32\log n$*. For any* $\gamma \geq 1$*, there is a* $\lambda > 0$ *such that if* (γ,λ)-ANN^n_X *can be solved by a deterministic or Las Vegas randomized cell-probing scheme on a table of* s *cells, each cell containing* w *bits, with expected* t *cell-probes for uniform and independent database and query, then* $t = \Omega\left(\frac{d}{\gamma^2\log\frac{sw\gamma^2}{nd}}\right)$ *or* $t = \Omega\left(\frac{nd}{\gamma^2(w+\log s)}\right)$*.*

Proof. Choose λ to satisfy $\gamma\lambda = \frac{d}{2} - \sqrt{2d\ln(2n)}$. Let μ be uniform distribution over X. We are going to show:

- $\mathbb{E}_{x\sim\mu}[\mu(N_{\gamma\lambda}(x))] \leq \frac{1}{2n}$;
- the λ-neighborhoods in X are (Φ, Ψ)-expanding under distribution μ for some $\Phi = 2^{\Omega(d/\gamma^2)}$ and $\Psi = 2^{\Omega(d/\gamma^2)}$.

Then the cell-probe lower bounds follows directly from Corollary 12.

First, by the Chernoff bound, $\mu(N_{\gamma\lambda}(x)) \leq \frac{1}{2n}$ for any point $x \in X$. Thus trivially $\mathbb{E}_{x\sim\mu}[\mu(N_{\gamma\lambda}(x))] \leq \frac{1}{2n}$.

On the other hand, for $d \geq 32\log n$ and n being sufficiently large, it holds that $\lambda \geq \frac{d}{4\gamma}$. Let $r = \frac{d}{2} - \frac{d}{8\gamma}$. And consider any $A \subseteq X$ with $\mu(A) \geq 2^{-(1-H(r/d))d}$. We have $|A| \geq 2^{dH(r/d)} \geq B(r)$. Then by Harper's theorem,

$$|N_\lambda(A)| \geq B(r+\lambda) \geq B\left(\frac{d}{2} + \frac{d}{8\gamma}\right) \geq 2^d - B\left(\frac{d}{2} - \frac{d}{8\gamma}\right) = 2^d - B(r) \geq 2^d - 2^{dH(r/d)},$$

which means $\mu(N_\lambda(A)) \geq 1 - 2^{-(1-H(r/d))d}$. In other words, the λ-neighborhoods in X are (Φ, Ψ)-expanding under distribution μ for $\Phi = \Psi = 2^{(1-H(r/d))d}$, where $r/d = \frac{1}{2} - \frac{1}{8\gamma}$. Apparently $1 - H(\frac{1}{2} - x) = \Theta(x^2)$ for small enough $x > 0$. Hence, $\Phi = \Psi = 2^{\Theta(d/\gamma^2)}$. ◀

5.2 Lower bound for ANN under L-infinity norm

Let $\Sigma = \{0, 1, \ldots, m\}$ and the metric space is $X = \Sigma^d$ with ℓ_∞ distance $\mathrm{dist}(x, y) = \|x - y\|_\infty$ for any $x, y \in X$.

Let μ be the distribution over X as defined in [2]: First define a distribution π over Σ as $p(i) = 2^{-(2\rho)^i}$ for all $i > 0$ and $\pi(0) = 1 - \sum_{i>0}\pi(i)$; and then μ is defined as $\mu(x_1, x_2, \ldots, x_d) = \pi(x_1)\pi(x_2)\ldots\pi(x_d)$.

The following isoperimetric inequality is proved in [2].

▶ **Lemma 15** (Lemma 9 of [2]). *For any $A \subseteq X$, it holds that $\mu(N_1(A)) \geq (\mu(A))^{1/\rho}$.*

Consider the (γ, λ)-approximate near-neighbor problem (γ, λ)-$\mathsf{ANN}^n_{\ell_\infty}$ defined in the metric space X under ℓ_∞ distance. The hard distribution for this problem is $\mu \times \mu^n$: For the database $y = (y_1, y_2, \ldots, y_n) \in X^n$, each database point y_i is sampled independently according to μ; and the query point x is sampled independently from X according to μ. The following lower bound has been proved in [2] and [12].

Fix any $\epsilon > 0$ and $0 < \delta < \frac{1}{2}$. Assume $\Omega\left(\log^{1+\epsilon} n\right) \leq d \leq o(n)$. For $3 < c \leq O(\log\log d)$, define $\rho = \frac{1}{2}(\frac{\epsilon}{4}\log d)^{1/c} > 10$. Now we choose $\gamma = \log_\rho \log d$ and $\lambda = 1$.

▶ **Theorem 16.** *With d, γ, λ, ρ and the metric space X defined as above, if (γ, λ)-$\mathsf{ANN}^n_{\ell_\infty}$ can be solved by a deterministic or Las Vegas randomized cell-probing scheme on a table of s cells, each cell containing $w \leq n^{1-2\delta}$ bits, with expected $t \leq \rho$ cell-probes under input distribution $\mu \times \mu^n$, then $sw = n^{\Omega(\rho/t)}$.*

Proof. The followings are true

- $\mu(N_{\gamma\lambda}(x)) = \frac{e^{-\log^{1+\epsilon/3} n}}{n} \leq \frac{1}{2n}$ for any $x \in X$ (Claim 6 in [2]);
- the λ-neighborhoods in X are $(n^{\delta\rho}, \frac{n^\delta}{n^\delta - 1})$-expanding under distribution μ for $\Phi = n^{\delta\rho}$ and $\Psi = 2^{\Omega(d/\gamma^2)}$.

To see the expansion is true, let $\Phi = n^{\delta\rho}$ and $\Psi = \frac{n^\delta}{n^\delta - 1}$. By Lemma 15, for any set $A \subset X$ with $\mu(A) \geq \Phi$, we have $\mu(N_\lambda(A)) \geq n^{-\delta} \geq 1 - \frac{1}{\Psi}$. This means λ-neighborhoods of $\mathcal{M}$ are $(n^{\delta\rho}, \frac{n^\delta}{n^\delta - 1})$-expanding.

Due to Corollary 12, either $\left(\frac{sw}{n^{1-\delta}}\right)^{O(t)} \geq n^{\delta\rho}$ or $= \Omega\left(\frac{n^{1-\delta}}{w + \log s}\right)$. The second bound is always higher with our ranges for w and t. The first bound gives $sw = n^{\Omega(\rho/t)}$. ◀

References

1 Amirali Abdullah and Suresh Venkatasubramanian. A directed isoperimetric inequality with application to bregman near neighbor lower bounds. In *STOC'15*.

2 Alexandr Andoni, Dorian Croitoru, and Mihai Pătraşcu. Hardness of nearest neighbor under L-infinity. In *FOCS'08*.

3 Alexandr Andoni, Piotr Indyk, and Mihai Pătraşcu. On the optimality of the dimensionality reduction method. In *FOCS'06*.

4 Alexandr Andoni and Ilya Razenshteyn. Optimal data-dependent hashing for approximate near neighbors. In *STOC'15*.

5 Omer Barkol and Yuval Rabani. Tighter lower bounds for nearest neighbor search and related problems in the cell probe model. *Journal of Computer and System Sciences*, 64(4):873–896, 2002. Conference version in *STOC'00*.

6 Allan Borodin, Rafail Ostrovsky, and Yuval Rabani. Lower bounds for high dimensional nearest neighbor search and related problems. In *Discrete and Computational Geometry*, pages 253–274, 2003. Conference version in *STOC'99*.

7 Amit Chakrabarti, Bernard Chazelle, Benjamin Gum, and Alexey Lvov. A lower bound on the complexity of approximate nearest-neighbor searching on the hamming cube. In *Discrete and Computational Geometry*, pages 313–328, 2003. Conference version in *STOC'99*.

8 Amit Chakrabarti and Oded Regev. An optimal randomised cell probe lower bound for approximate nearest neighbour searching. In *SIAM Journal on Computing*, 39(5):1919–1940,2010. Conference version in *FOCS'04*.

9 L.H. Harper. Optimal numberings and isoperimetric problems on graphs. *Journal of Combinatorial Theory*, 1(3):385 – 393, 1966.

10 Piotr Indyk. Nearest neighbors in high-dimensional spaces. *Handbook of Discrete and Computational Geometry*, pages 877–892, 2004.

11 T.S. Jayram, Subhash Khot, Ravi Kumar, and Yuval Rabani. Cell-probe lower bounds for the partial match problem. In *Journal of Computer and System Sciences*, 69(3):435–447, 2004. Conference version in *STOC'03*.

12 Michael Kapralov and Rina Panigrahy. NNS lower bounds via metric expansion for ℓ_∞ and EMD. In *ICALP'12*.

13 Kasper Green Larsen. Higher cell probe lower bounds for evaluating polynomials. In *FOCS'12*.

14 Ding Liu. A strong lower bound for approximate nearest neighbor searching. *Information Processing Letters*, 92(1):23–29, 2004.

15 Peter Bro Miltersen, Noam Nisan, Shmuel Safra, and Avi Wigderson. On data structures and asymmetric communication complexity. *Journal of Computer and System Sciences*, 57(1):37–49, 1998. Conference version in *STOC'95*.

16 Rina Panigrahy, Kunal Talwar, and Udi Wieder. A geometric approach to lower bounds for approximate near-neighbor search and partial match. In *FOCS'08*.

17 Rina Panigrahy, Kunal Talwar, and Udi Wieder. Lower bounds on near neighbor search via metric expansion. In *FOCS'10*.

18 Mihai Pătraşcu and Mikkel Thorup. Higher lower bounds for near-neighbor and further rich problems. *SIAM Journal on Computing*, 39(2):730–741, 2010. Conference version in *FOCS'06*.

19 Alan Siegel. On universal classes of fast high performance hash functions, their time-space tradeoff, and their applications. In *FOCS'89*.

20 Yaoyu Wang and Yitong Yin. Certificates in data structures. In *ICALP'14*.

21 Yitong Yin. Simple average-case lower bounds for approximate near-neighbor from isoperimetric inequalities. *arXiv preprint* arXiv:1602.05391.